Ch 4. Introduction to Recursion

Recursive functions

- A recursive function
  - A function that calls itself, either directly, or indirectly (through another function).
- Example: factorial

  Let $f(n) = n! = n * (n - 1) * (n - 2) * \cdots * 1$
- A recursive definition of the factorial function

  $f(0) = 0! = 1$

  $f(1) = 1! = 1$

  $f(n) = n! = n * (n - 1)! = f(n - 1)$

Iterative Function Factorial

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}

int f( int n )
{
   int result = 1;
   for( int i = 2; i <= n; i++ )
      result *= i;
   return result;
}
```

$$f(n) = n! = n * (n - 1) * (n - 2) * \cdots * 1$$

Iterative Function Factorial

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}

int f( int n )
{
   int result = 1;
   for( int i = n; i >= 2; i-- )
      result *= i;
   return result;
}
```

$$f(n) = n! = n * (n - 1) * (n - 2) * \cdots * 1$$

Recursive Function Factorial

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return                ;
}
```

$f(n) = 1 \qquad \text{for } n \leq 1 \quad (n! = 1)$

$f(n) = n * f(n-1) \quad \text{for } n > 1 \quad (n! = n * (n-1)!)$

Recursive Function Factorial

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```

$$f(n) = 1 \qquad \text{for } n \leq 1 \quad (n! = 1)$$

$$f(n) = n * f(\, n - 1\,) \qquad \text{for } n > 1 \quad (n! = n * (\, n - 1\,)!\,)$$

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}
```

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else
       return n * f( n - 1 );
}
```

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}
```

2

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else
       return n * f( n - 1 );
}
```

1

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}
```

3

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else
       return n * f( n - 1 );
}
```

2

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}
```

4

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else
       return n * f( n - 1 );
}
```

3

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << n << "! = " << f( n ) << endl;
}
```

5

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else
       return n * f( n - 1 );
}
```

4

```cpp
#include <iostream>
using namespace std;

int f( int n );

int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```

4

```
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
```

```
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }
```

```
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }
```

```
         int f( int n ) {
            if( n <= 1 ) return 1;
            else
               return n * f( n - 1 ); }
```

```
                4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                        3


   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }


      int f( int n ) {
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }


         int f( int n ) {
            if( n <= 1 ) return 1;
            else
               return n * f( n - 1 ); }
```

```
                  4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                        3


                  3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }
                           2


                     2
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }


         int f( int n ) {
            if( n <= 1 ) return 1;
            else
               return n * f( n - 1 ); }
```

```
                  4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                        3


                  3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }
                           2


                  2
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }
                              1


                  1
         int f( int n ) {
            if( n <= 1 ) return 1;
            else
               return n * f( n - 1 ); }
```

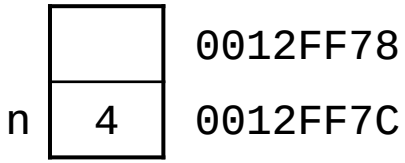
```
                4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                        3


               3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }
                           2


            2
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }
                              1


         1
      int f( int n ) {          1
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }
```

```
                4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                        3


               3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
         return n * f( n - 1 ); }
                           2


            2
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
                  2
            return n * f( n - 1 ); }
                     1    1


         1
      int f( int n ) {          1
         if( n <= 1 ) return 1;
         else
            return n * f( n - 1 ); }
```

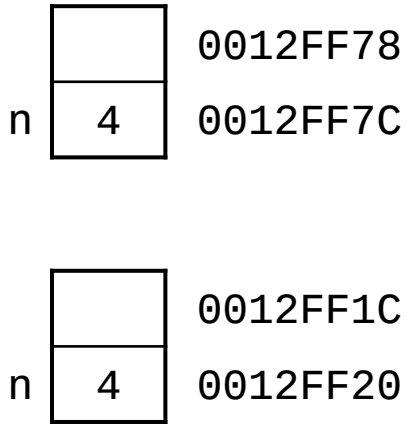
```
                4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
      return n * f( n - 1 ); }
                       3


               3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
                  6
         return n * f( n - 1 ); }
                        2   2


            2
         int f( int n ) {
            if( n <= 1 ) return 1;
            else
                     2
               return n * f( n - 1 ); }
                          1   1


         1
            int f( int n ) {        1
               if( n <= 1 ) return 1;
               else
                  return n * f( n - 1 ); }
```

```
                4
int f( int n ) {
   if( n <= 1 ) return 1;
   else
                24
      return n * f( n - 1 ); }
                   6   3


                3
   int f( int n ) {
      if( n <= 1 ) return 1;
      else
                   6
         return n * f( n - 1 ); }
                      2   2


                2
      int f( int n ) {
         if( n <= 1 ) return 1;
         else
                      2
            return n * f( n - 1 ); }
                         1   1


                1
         int f( int n ) {          1
            if( n <= 1 ) return 1;
            else
               return n * f( n - 1 ); }
```

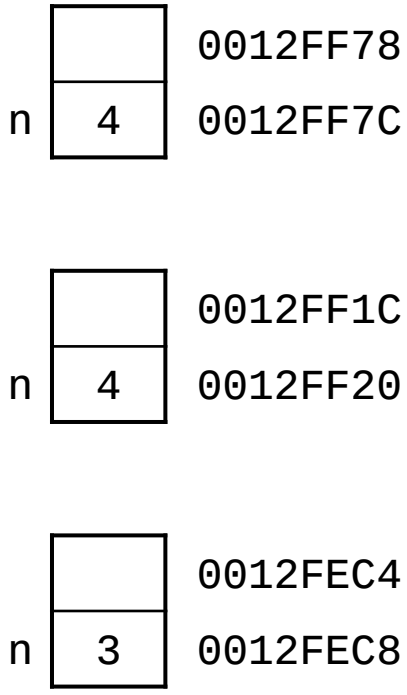
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



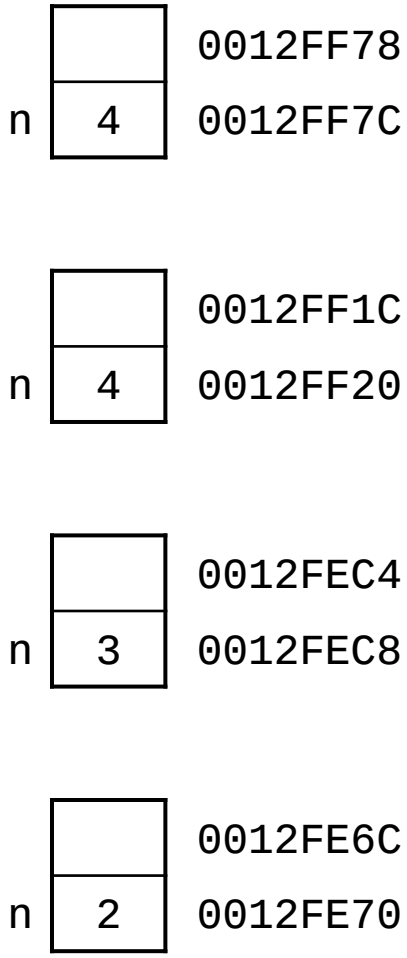
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



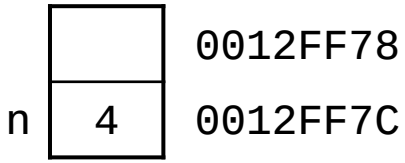
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



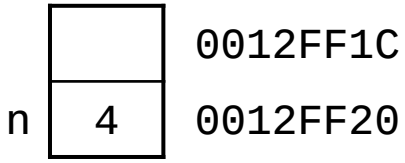
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



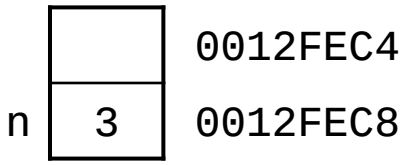
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}
```



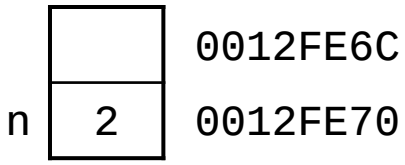

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



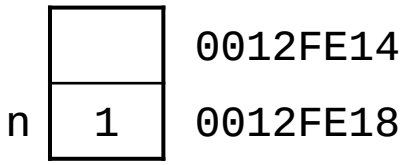

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



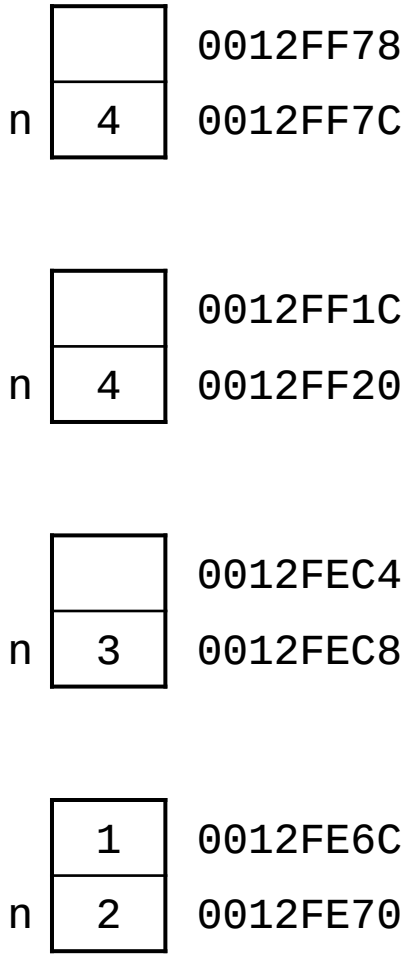

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



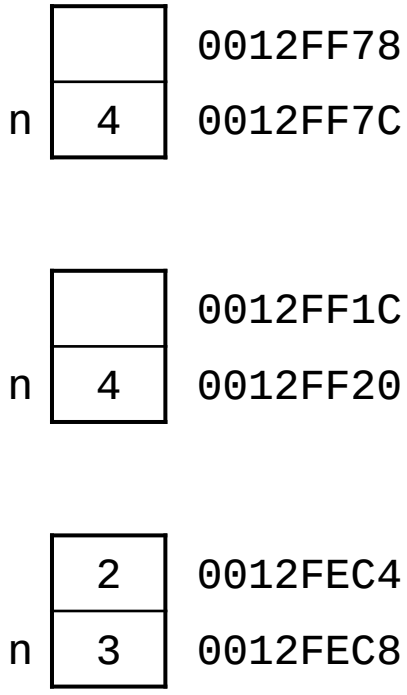

```
int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



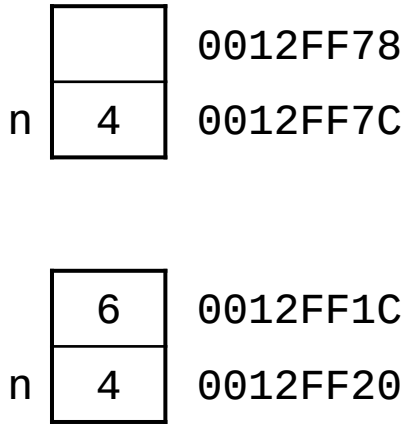
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



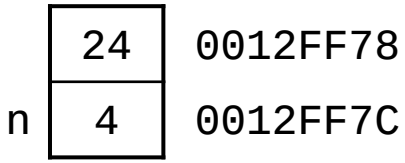
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



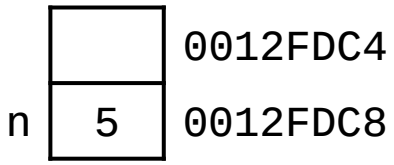
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



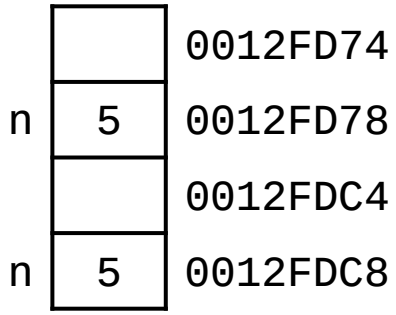
```
int main()
{
   int n = 4;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}

int f( int n )
{
   if( n <= 1 ) return 1;
   else return n * f( n - 1 );
}
```



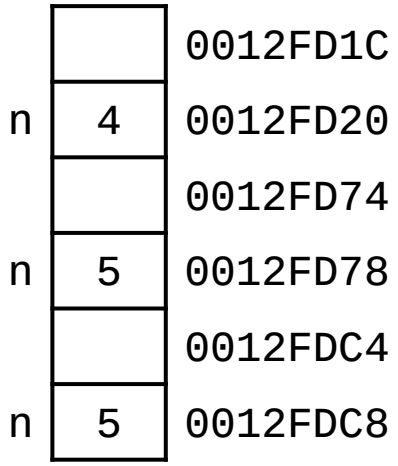
```cpp
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



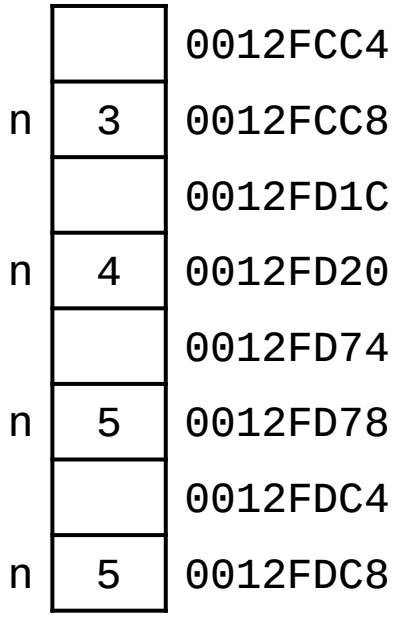
```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



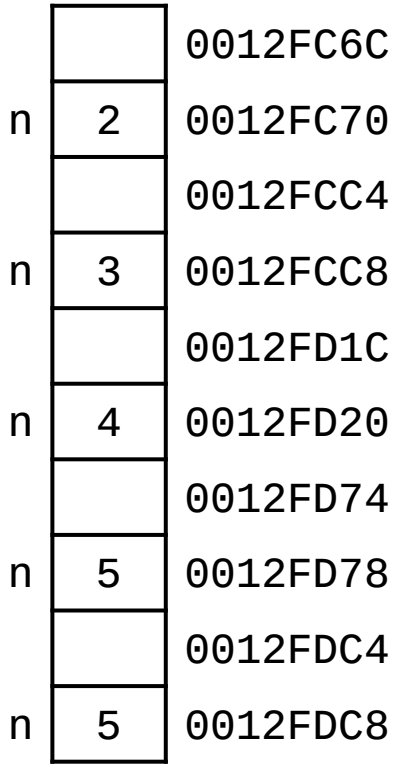
```cpp
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



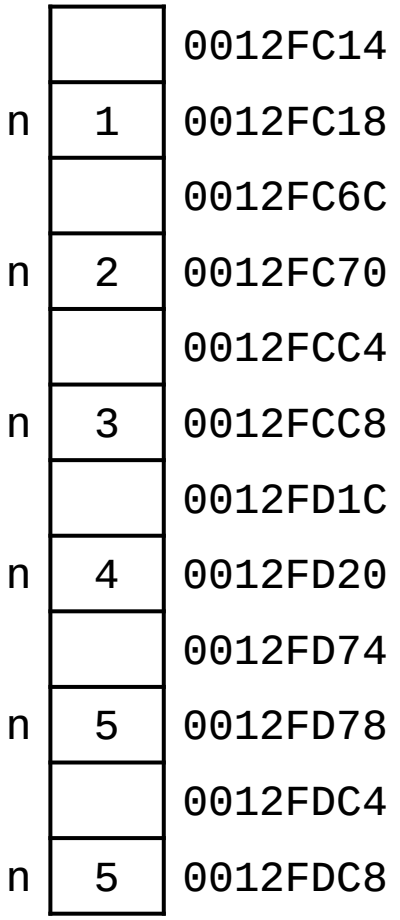
```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



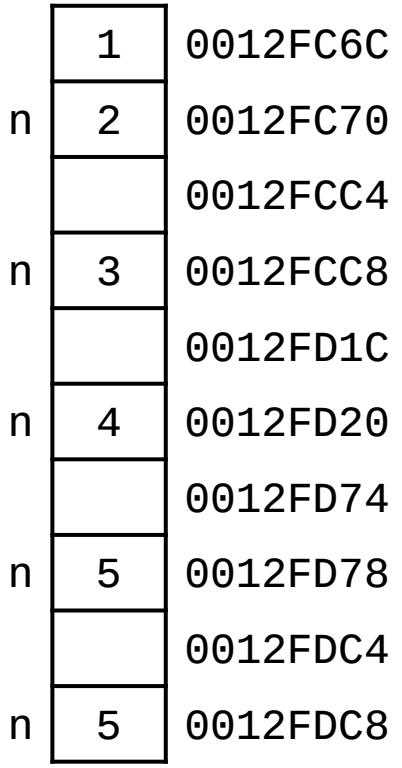
```cpp
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



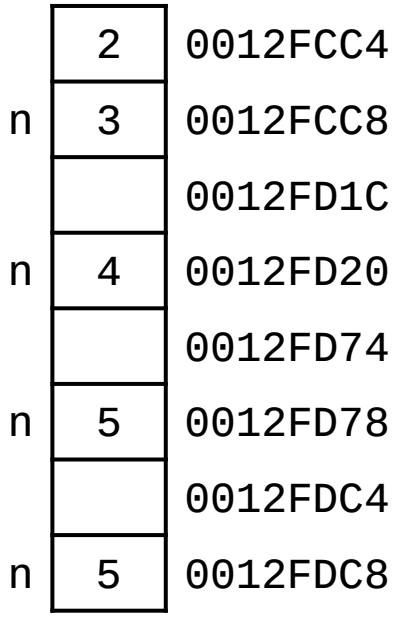
```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



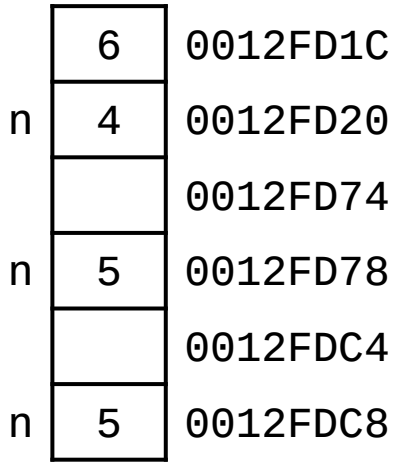
```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



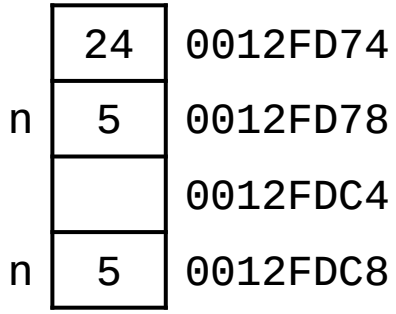
```cpp
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



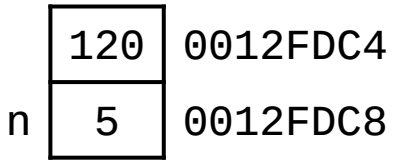
```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



```cpp
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



```
#include <iostream>
#include <iomanip>
using namespace std;

int f( int );

int main()
{
   int n = 5;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return 1;
   else
      return n * f( n - 1 );
}
```



Recursive Fibonacci Function

Let $f(n)$ be the n-th fibonacci number

$f(n) = n$ for $n \leq 1$ ($f(0) = 0, f(1) = 1$)

$f(n) = f(n-1) + f(n-2)$ for $n > 1$

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << "f( " << n << " ) = " << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return  ;
   else
      return                          ;
}
```

Recursive Fibonacci Function

Let $f(n)$ be the n-th fibonacci number

$f(n) = n$ for $n \leq 1$ ($f(0) = 0, f(1) = 1$)

$f(n) = f(n-1) + f(n-2)$ for $n > 1$

```
int main()
{
   int n;
   cout << "Enter a positive Integer: ";
   cin >> n;
   cout << "f( " << n << " ) = " << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

1 0

1 0

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

3

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

1 1

2 1

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

2 1

3 2

```
int main()
{
   int n;
   cout << "Enter a nonnegative Integer: ";
   cin >> n;
   cout << f( n ) << endl;
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

3 2

4 3

3

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

0

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1 0 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

1 2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1 0 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

1 2 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1 0 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

3

1 2

1 1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

2

1 1

0 0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

1

1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

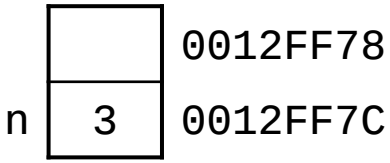
1

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

0

```
int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f(n-1)+f(n-2);
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```

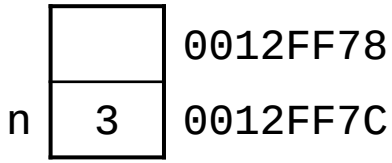



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

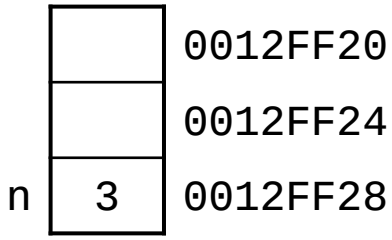
```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```



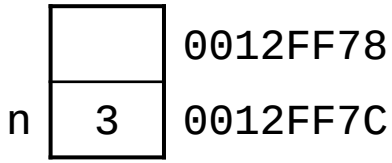

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



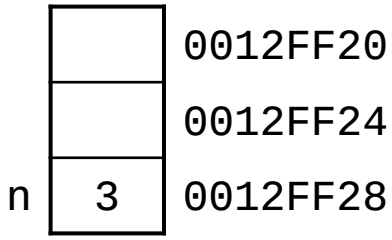

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

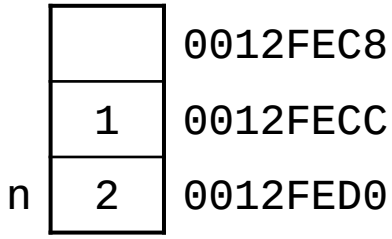
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```




```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

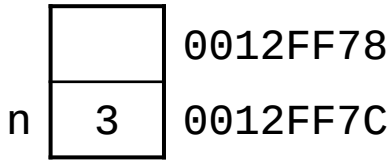



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

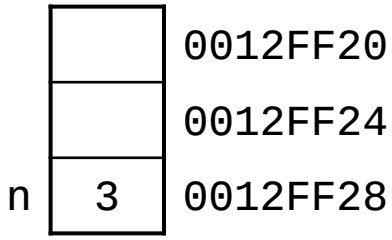



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

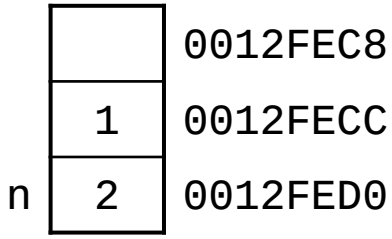
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```



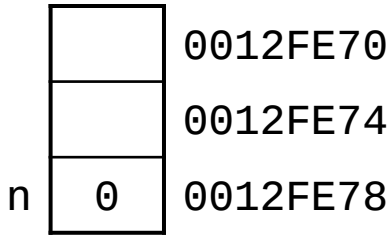

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



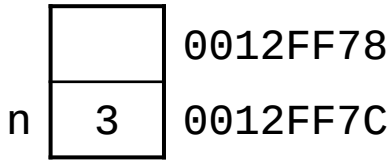

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



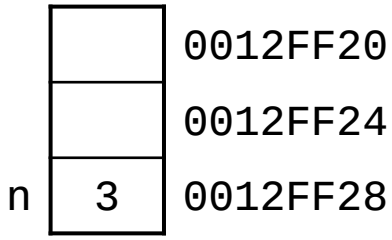

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



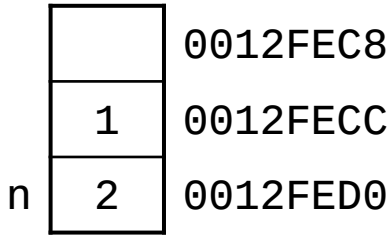
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```




```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

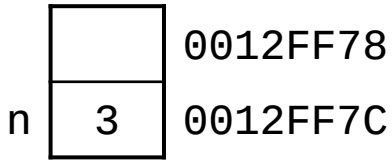



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

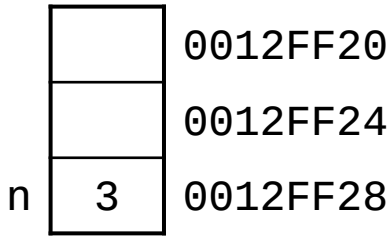



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

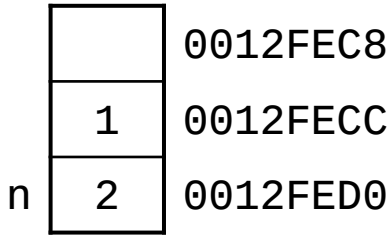
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```



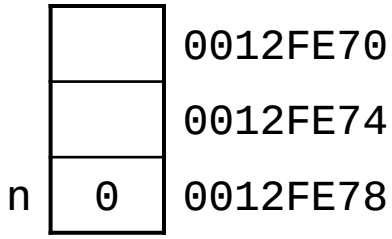

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



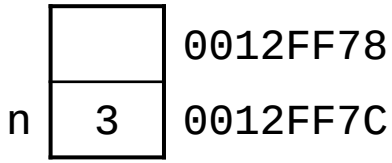

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



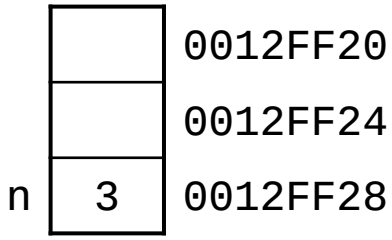

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



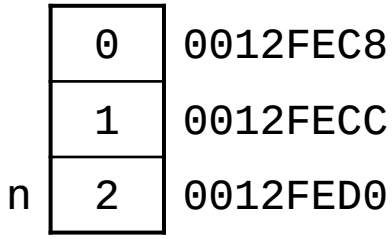
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```




```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

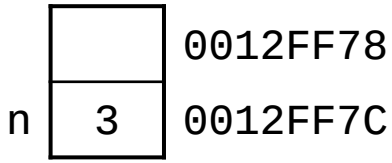



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

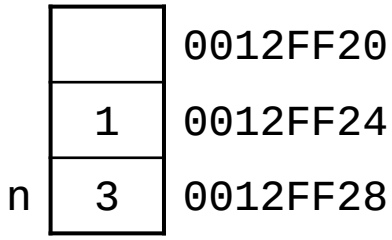



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```



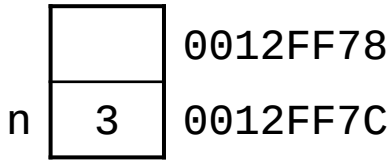

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```



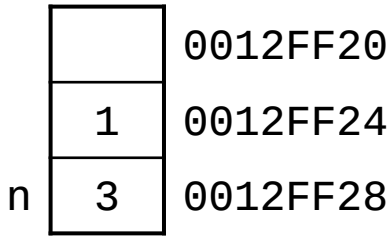

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

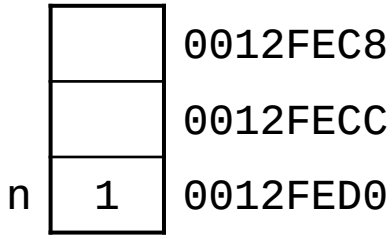
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```




```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

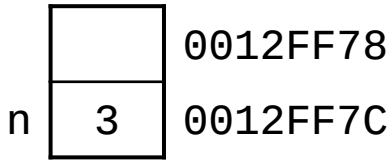



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

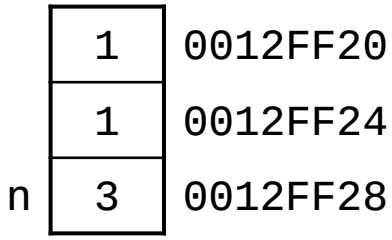



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```



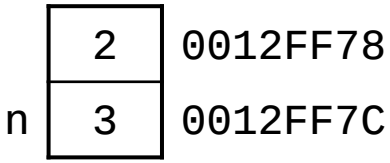

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```




```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}
```

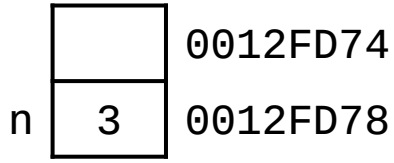



```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

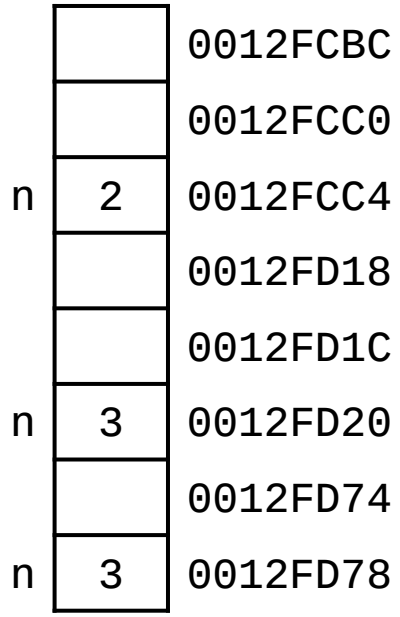
```
int f( int n )
{
   if( n <= 1 ) return n;
   else return f( n - 1 ) + f( n - 2 );
}
```

```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

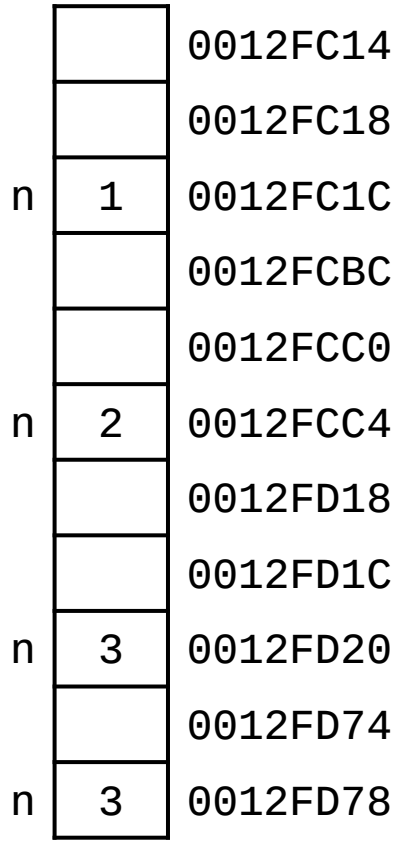


```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```

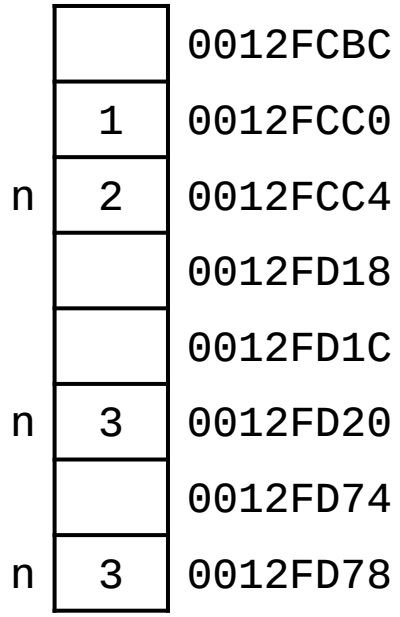
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



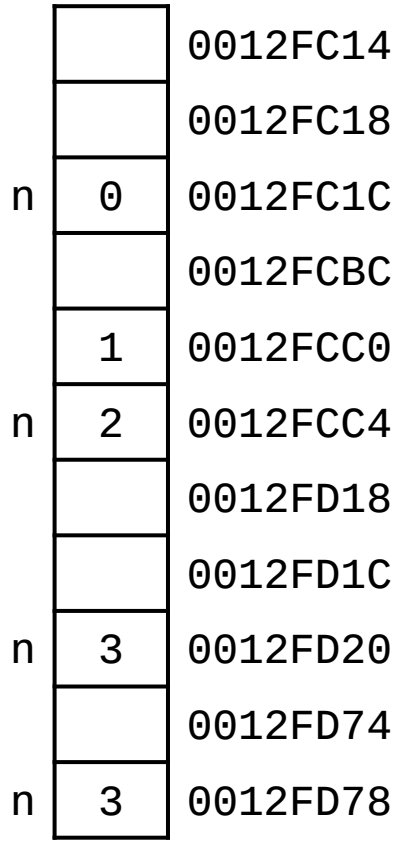
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



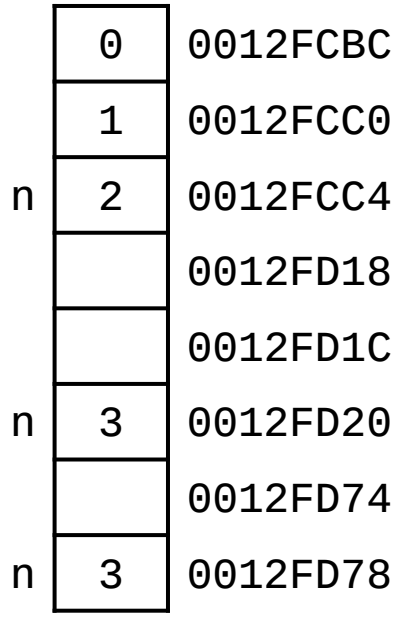
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



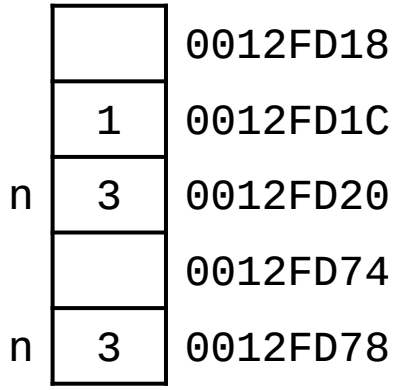
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



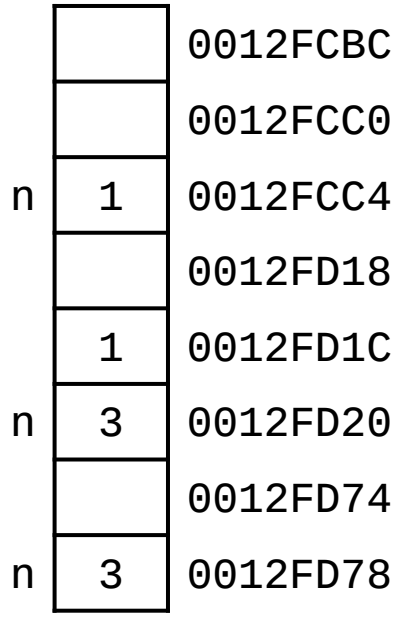
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



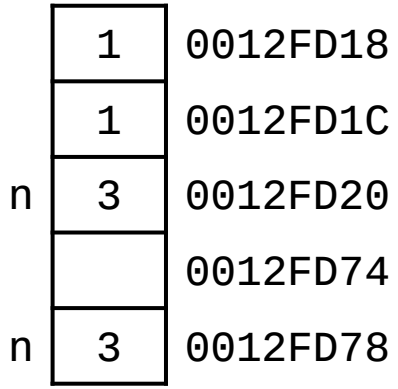
```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```



```
int main()
{
   int n = 3;
   cout << f( n ) << endl;
}

int f( int n )
{
   if( n <= 1 )
      return n;
   else
      return f( n - 1 ) + f( n - 2 );
}
```